

Claims

The following is a copy of Applicant's claims that identifies language being added with underlining ("___") and language being deleted with strikethrough ("—"), as is applicable:

1. (Currently amended) A method for dynamically patching code, comprising the steps of:

intercepting original program instructions during execution of the program;

determining if a an original program instruction is to be replaced; and

dynamically replacing the original program instruction with a replacement instruction by fetching the replacement instruction and storing the replacement instruction in a code cache from which the replacement instruction can be executed in lieu of the original program instruction.

2. Canceled.

3. (Currently amended) The method of claim 2 1, wherein ~~the step of~~ dynamically replacing the original program instruction further comprises executing the replacement instruction in lieu of the original program instruction each time a function associated with the original program instruction is required.

4. (Currently amended) The method of claim 3 1, wherein the replacement instruction comprises part of a patch that is made available via an application programming interface.

5. (Currently amended) The method of claim 1, further comprising ~~the step of~~, prior to determining if a an original program instruction is to be replaced, determining if the original program instruction has been cached.

6. (Currently amended) The method of claim 5, further comprising ~~the step of~~ executing the cached instruction in lieu of the original program instruction if an associated instruction has been cached.

7. (Currently amended) The method of claim 1, further comprising ~~the step of~~, prior to intercepting program instructions, gaining control over execution of original program instructions by injecting a ~~dynamic execution layer~~ software interface into the program.

8. (Currently amended) The method of claim 1, further comprising ~~the step of~~ dynamically receiving information about original program instructions to be replaced and replacement instructions to replace the original program instructions.

9. (Currently amended) The method of claim 1, further comprising ~~the step of~~ executing transition code if a an original program instruction to be replaced is currently running.

10. (Currently amended) A system for dynamically patching code, comprising:

means for gaining control over execution of a program;

means for intercepting original program instructions during execution of the program;

means for determining if a an original program instruction is to be replaced with a new instruction; and

means for dynamically replacing the original program instruction with a replacement instruction, the means for dynamically replacing being configured to fetch the replacement instruction and store the replacement instruction in a code cache from which the replacement instruction can be executed in lieu of the original program instruction.

11. Canceled.

12. (Currently amended) The system of claim 10, further comprising means for determining if a an original program instruction has been cached.

13. (Currently amended) The system of claim 10, further comprising means for dynamically receiving information about original program instructions to be replaced and replacement instructions that are configured to replace the original program instructions.

14. (Currently amended) A dynamic patching program stored on a computer-readable medium, the program comprising:

logic configured to gain control over execution of a program;

logic configured to intercept original program instructions during program execution;

logic configured to determine if a an original program instruction is to be replaced; and

logic configured to dynamically replace the program instruction with a replacement instruction by fetching the replacement instruction and storing the replacement instruction in a code cache from which the replacement instruction can be executed in lieu of the original program instruction.

15. Canceled.

16. (Currently amended) The program of claim 14, further comprising logic configured to determine if a an original program instruction has been cached.

17. (Currently amended) The program of claim 14, further comprising logic configured to dynamically receive information about original program instructions to be replaced and replacement instructions that are configured to replace the original program instructions.

18. (Currently amended) A method for dynamically patching code, comprising ~~the steps of~~:

gaining control over the execution of a program using a software interface;

intercepting original program instructions during execution of the program;

determining whether the original program instructions have been cached in a code cache of the software interface and, if so, executing the cached instructions;

if the original program instructions have not been cached, determining if the original program instructions are to be replaced; ~~and~~

dynamically replacing the original program instructions with replacement instructions if it is determined that the original program instructions are to be replaced by fetching the replacement instructions and storing the replacement instructions in the code cache of the software interface; and

executing the replacement instructions from the code cache of the software interface in lieu of the original program instructions.

19. Canceled.

20. (Currently amended) The method of claim ~~18,19, wherein the step of~~ dynamically replacing the program instructions further comprises comprising executing the replacement instructions in lieu of the program instructions each time a functionality associated with the original program instructions is required.

21. (Original) The method of claim 20, wherein the replacement instructions comprise part of a patch that is made available via an application programming interface.

22. (Currently amended) The method of claim 18, further comprising ~~the~~
~~step of~~ executing transition code if ~~a~~ an original program instruction to be replaced is
currently running.